

RECEIVED
CENTRAL FAX CENTER

OCT 31 2006

REMARKS

5 Claims 3-9, 12-13, 15-18 and 20-22 are hereby amended for clarification requested by Supervisory Examiner Corrielus. There are still 22 claims in this application.

Reconsideration, withdrawal of the final rejection and continued examination is respectfully requested. Applicant is grateful that examiners Ly and Corrielus requested and held numerous

10 lengthy interviews with the Applicant's attorney in order to expedite prosecution and allow this application. Interviews were held from 7/19/2006 to 7/24/2006 and several claims' amendments were proposed by the applicant according to examiner Corrielus' comments, in the Proposed Amendment of 7/23/2006. However, a final office action was sent instead which is an exact repetition of the previous office action and fails to address the claims' amendments and the

15 interviews' topic and purpose. Furthermore, at the request, Applicant provided Examiner Ly with a written authorization to send proposed amendments via e-mail. Applicant proposed an amendment to Specification and a written clarification of claim 3 and specification in several e-mails but there was no written response from the USPTO. There was assurance from both examiners that they had no further problems and will allow the application, stated on 8/16/2006

20 and 8/17/2006. Instead, on 10/20/2006 Applicant was informed by Supervisory Examiner Breene that he is now in charge since the Supervisory Examiner Corrielus is off this case so Applicant forwarded all e-mails from August 2006 to Mr. Breene, on 10/23/2006, and he promised to review the case before the 3-month deadline. However, Applicant was advised on 10/30/2006 that only an official amendment will be considered. This was confirmed by Examiner Ly who

25 advised Applicant that the only problem he has is as written by examiner Corrielus, and that he will do another review and make a decision after receipt of the official amendment after final. Neither examiner would hold an examiner's interview or propose an Examiner's Amendment, although that would expedite the prosecution.

30 As repeated by Applicant numerous times during interviews and in e-mails, the invention has not been properly understood. Thus claims and specification were amended for clarification,

requested in the Continuation of Substance of Interview by Primary Examiner Corrielus on 7/31/2006 which mentions the Proposed Amendment of 7/23/2006 and states that:

Continuation of Substance of Interview including description of the general nature of what was agreed to if an agreement was reached, or any other comments: After further review the invention as claimed, it appears that the language in the proposed amendment filed by the Applicant on July 23, 2006 (see attached) is not supported by the original specification. The examiner has carefully reviewed the specification with respect to the proposed claimed invention. For example, claim 3 requires that the pointer to the function to call to process the operation specified by the opcode is replaced by a pointer to an intermediate processing function and auxiliary data structure, wherein the auxiliary data structure includes a pointer to store the pointer the function to call to process the operation specified by the Opcode (see specification page 7, lines 13-17), which is different from the proposed claimed language. Applicant argues that such proposed language is supported by the specification. If the Applicant believes that the proposed amendment in compliant with the 112 first paragraph of enablement, the Applicant is welcome to submit such amendment and point exactly where such amendment is support by the specification.

15 First element of claim 3, as amended, states: "augmenting said first operation code in the access plan with a replacement pointer to an intermediate function, ~~said intermediate function~~ including having a data structure.

Specification paragraph on p. 7, li. 11-15, as amended, states the same thing: "Instead of using the pointer to the function to call to process the operation specified by the OPCODE is replaced by in this aspect the OPCODE is augmented with a pointer to the intermediate processing function and an having an auxiliary data structure.

Second element of claim 3, as amended, states:

25 and for storing a pointer to said executable function in the data structure.

Specification paragraph on p. 7, li. 13-17, as amended, states the same thing: the auxiliary data structure includes a pointer to store the pointer the function to call to process the operation specified by the OPCODE.

30 Because the only problem with the Proposed Amendment of 7/23/2006 was described by examiner Corrielus and it is an 112 objection, clarified with this amendment to the specification, a reconsideration is respectfully requested because all claims are in good condition for allowance after the amendments. None of the cited references discloses the subject matter and features of

claims 1-22 of the present invention and, even if they did show some individual features, they would not be able to meet the claims of the present invention which provide new and unexpected results over these references and are thus not anticipated under Section 102 and unobvious and patentable under Section 103.

5

Response to Argument section of the Final Office Action, p. 2-3, is a repetition of the rejections and further failed to clarify the rejections and prior art. It is vague and broad and failed to directly respond to Applicant's arguments from the Response to the first office action after appeal, dated 5/8/2006, to clearly point out where in Chow reference are found which elements 10 used for 102 and 103 rejections. Moreover, it did not interpret the claims in light of the specification and drawings. Thus, the Final Office Action, as a whole, failed to address Applicant's numerous arguments about apparent misunderstanding of the present invention and prior art. It also failed to clarify ambiguities regarding sited elements of prior art, such as which 15 claimed element of the present invention is allegedly taught by which prior art element, as pointed out in the Response.

Because the Final Office Action is moot regarding Applicant's remarks from p. 2-4 of the Response of 5/8/2006, they are incorporated herein by reference and Applicant respectfully requests an explanation as to the rules from 35 USC and 37 CFR allowing such actions.

20

3-6. Claims 1-22 are pending. Claims 1-8, 10-17 and 19-22 stand rejected under 35 U.S.C. 102(e) as being anticipated by US Patent No. 5,875,334 issued to Chow et al. ("Chow").

Although the reference has the common assignee it cannot be used to reject these claims as 25 shown below and no action is needed under 6.

With respect to claim 1, Office Action stated: "Chow teaches a method for pre-processing an access plan generated for a query in a relational database management system to include said access plan including a plurality of operation codes, each of said operation codes being 30 associated with one or more executable functions for performing the query (figs. 1-3,

Pre-processing a SQL query statement is processed through a query compiler for generating a access executable plan for a database system such as IBM's DB2; this process requiring a run-time interpreter for looking up local variable; also this executable plan includes a plurality of SOD Query control statements, which are parsing into some operation codes into query graph

- 5 model (QGM) representation of the statements, from which they are then processed to optimized QGM, an access execution plan such as cost-based optimized access plan and they are to be produced operation code or opcode by code generation module (item 118): abstract, col. 1, lines 15-35, col. 6, lines 25-67 and col. 8, lines 8-58; also col. 10, lines 1-40); determining from the access plan an executable function associated with a first operation code (fig. 1, parser, the
- 10 optimized QGM, and threaded code generation are used to produce operation code of SQL query statements to get the optimized access plan or access execution plan : figs. 1 and 2, item 115 & 118, col. 6, lines 25-67 and col. 8, lines 8-58, col. 14, lines 40-67 and col. 15, lines 1-67 and col. 16, lines 1-67); and augmenting said first operation code in the access plan with a pointer to said executable function to provide a direct call mechanism replacing lookup function of a runtime
- 15 interpreter (figs. 1 & 2, the extracted SQL statement or control statements is used to produce opcode based on parser, QGM and code generation and replacing this opcode (fig.2 and col. 15, lines 45-67 and col. 17, lines 25-67); the executable function in the run-time interpreter with a compiler generated local variables storing in symbol table for looking up with referencing local variables or pointers or indexes: fig. 4, Col. 24, lines 36-67 and col. 25, lines 1-8; also, col. 15, lines 1-67, col. 17, lines 1-28, Col. 20, lines 1-45 and col. 21, lines 5-55)."

The Applicant respectfully objects to the misinterpretation that Chow teaches "a method for pre-processing an access plan generated for a query in a relational database management system to include said access plan including a plurality of operation codes, each of said operation codes being associated with one or more executable functions for performing the query" because Chow does not have the quoted language. It is the language taken verbatim from the preamble of claim 1 of the present invention. Moreover, this language failed to include the language from lines 2-3 of claim 1 of the present invention, namely, "direct call replacing a lookup function of the run time interpreter".

All independent claims 1, 10 and 19 of the present invention are specifically directed to show an improvement of a standard database management system which includes implementation of a direct call mechanism replacing the lookup function of a run-time interpreter and a method for pre-processing an already created access plan to provide a direct call mechanism in such a system. They recite novel structure and thus distinguish over the cited prior art, under 35 U.S.C. 5 102(e) and 103(a). This is described in Title, Figs. 1-4 and Specification on p. 2, li. 16-31; p. 3, li. 2-10, p. 6, li. 14-30, pages 7-9, and is used to provide faster access which is cost-effective.

Specifically, the independent claim 1 states:

10 1. A method for **pre-processing an access plan** generated for a query in a relational database management system to include a **direct call mechanism replacing a lookup function of a run-time interpreter**, said access plan including a plurality of operation codes, each of said operation codes being associated with one or more executable functions for performing the query, said method comprising the steps of:

15 (a) determining from the access plan an executable function associated with a first operation code; and

 (b) augmenting said first operation code in the access plan with a pointer to said executable function to provide a direct call mechanism replacing a lookup function of a run-time interpreter.

20

As can be see from the bolded terms, claimed method is for **pre-processing of an existing access plan** which has op. codes and is to be interpreted by a Software Interpreter (not a Compiler). Step (a) determines, for an op. code from the existing access plan, an associated executable function which should be interpreted if there is no present invention. Step (b) augments the op. code, inside the existing access plan, and disallows interpretation of the executable function by the Interpreter because it replaces the op. code with a **direct pointer to the function**, the function being located outside the access plan, which implements the operation indicated by the op. code, thus removing the interpretive step. This means that later on, when the access plan is being executed at run-time, instead of interpreting the op. code, the pointer is used to call the replacement code, thus a term "direct call", which is described in much more detail in the Specification.

Thus, the claims are directed to preprocessing, which happens prior to execution, which will allow a run-time improvement of execution of an existing and previously optimized access plan whose steps have already been determined and are not changed by the present invention but are

5 substituted with pointers (Spec. p. 7, li. 5-10) in order to increase the run-time speed. As can be seen on p. 7, li. 25, the pre-processing is performed by the Access Plan Manager prior to storing the improved access path plan into a memory cache and before the execution. As can be seen on p. 8, li. 14-16, this processed code section has pointers and is ready for execution, described on p. 8, li. 17-29.

10

Applicant respectfully objects to the very lengthy and numerous paragraphs, sometimes over a column long, used to reject claims of the present invention because the Office Action lacks the required specificity of line number or element number and a corresponding element in the rejected claims of the present invention, thus making rejections overly broad and vague and, 15 thus, weak. It is, therefore, respectfully requested that a more specific and exact references be used in the future. This kind of citation is not permitting a direct argument on point for each rejection. Moreover, there are many extraneous elements from the Chow reference in the Office Action that have nothing to do with the claimed invention which are cluttering the Office Action and showing lack of understanding of both the present invention and the reference.

20

It is true that Chow reference mentions access plan, interpreter, data structures, etc. These components are mentioned in most database applications. However, Chow reference is from a different field and performs a different function. It is directed to a method used to extend a SQL compiler for handling control statements packaged with query statements (Title, Specification).

25 As Figs. 1-6 show, the end result of this reference is an access plan ("executable plan") and it teaches compilation of query statements (col. 1-8, col. 9 li. 1-21). Control statements are WHILE and REPEAT statements which are performed as a loop (col. 10, li. 12-18, col. 12, li 10-21, col. 17, li. 1-28). Sections in col. 14, li. 37-67 to col. 15, li. 1-67, col. 17 li. 29-67, col. 18, li. 13-39 describe a Pre-processing. However, here it is pre-processing of a skeleton of control statements 30 which is performed before compilation and before an access plan is created (specifically, in col.

18, li. 13-24). A lookup table in this reference is a symbol lookup table which stores variables and not op. code, as shown in numerous places such as, col. 18, li. 37-39, col. 20, li. 1-47, col. 24, li. 36-53. Sections of cols. 19-25 describe the work of a parser and compiler. The parser creates a syntax tree AST whose nodes are accessed by node pointers and stored in a stack. Col. 5 25, li. 1-8 describes that variables are indexed to maps and stored in data structures. Pointers to the variable data structure are stored in a symbol table entry.

Regarding preamble, it is noted with appreciation that Office Action itself stated, on top of p. 4, that Chow is directed to a pre-processing of a query, by a compiler, for generating an executable 10 plan. Moreover, it states that interpreter there is looking up variables. Therefore, Office Action itself held that Chow does not teach pre-processing of an access plan but of a query, before an access plan is even made, and a direct call is not even mentioned. Furthermore, element 118 of Chow is a threaded code generator and not an op. code generator and abstract, col. 1, col. 6, col. 8 and col. 10 citations are not relevant to the present invention. Therefore, the preamble of the 15 independent claims 1, 10 and 19 of the present invention is not anticipated by Chow.

Regarding bottom of p. 4 of Office action, it states: "determining from the access plan an executable function associated with a first operation code (fig. 1, parser, the optimized QGM, and threaded code generation are used to produce operation code of SQL query statements to get 20 the optimized access plan or access execution plan : figs. 1 and 2, item 115 & 118, col. 6, lines 25-67 and col. 8, lines 8-58, col. 14, lines 40-67 and col. 15, lines 1-67 and col. 16, lines 1-67);".

This language is a misrepresentation because it is taken from element (a) of independent claims of the present invention and it cannot be found in Chow reference. Chow is not determining 25 anything from an access plan because its final product is an access plan, as shown above. As also shown above, cited sections are teaching use of a parser and compiler, before an access plan is created. Therefore, Chow does not teach an analysis of an existing access plan and determining from it an executable function associated with an op. code, as claimed in element (a) of the present invention.

Regarding bottom of p. 4 of Office action, it states: "augmenting said first operation code in the access plan with a pointer to said executable function to provide a direct call mechanism replacing lookup function of a runtime interpreter (figs. 1 & 2, the extracted SQL statement or control statements is used to produce opcode based on parser, QGM and code generation and 5 replacing this opcode (fig.2 and col. 15, lines 45-67 and col. 17, lines 25-67); the executable function in the run-time interpreter with a compiler generated local variables storing in symbol table for looking up with referencing local variables or pointers or indexes: fig. 4, Col. 24, lines 36-67 and col. 25, lines 1-8; also, col. 15, lines 1-67, col. 17, lines 1-28, Col. 20, lines 1-45 and col. 21, lines 5-55)."

10

This language is a misrepresentation because it is taken from element (b) of independent claims of the present invention and it cannot be found in Chow reference. As discussed above, Chow does not even mention augmenting or replacing op. codes in the access plan nor the direct call mechanism. It is noted with appreciation that Office Action on top of p. 5 held that Chow teaches 15 that a look up is connected with a symbol table which references local variables. Therefore, Chow does not teach augmenting an op. code in the access plan with a pointer to an executable function, as claimed in element (b) of the present invention but a variable symbol table. Therefore, the elements (a) and (b) of the independent claims 1, 10 and 19 of the present invention are also not anticipated by Chow.

20

Moreover, Chow reference does not have any and all elements from the independent claims of the present invention. It does not pre-process an existing access plan, does not use a direct call mechanism, does not replace a lookup function of a run-time interpreter, does not include pointers to functions, does not use the normal executable function from an already generated 25 access plan (step (a)), does not create a new access plan with pointers to the same executable function (step (b)), etc.

Therefore, Chow reference clearly does not teach and is not directed to a pre-processing method replacing a lookup function of a run-time interpreter, which is claimed in the present invention, 30 because these and other features of the present invention, recited in the claims and throughout the

Specification, are not shown in this reference. Moreover, the cited reference does not show any features of the present invention, operating in the same way and for the same purpose. Thus, the reference is from a different art field.

- 5 To establish *prima facie* case of anticipation of a claimed invention, the standard for determining novelty under 35 USC 102 was been set forth by the courts *In re Bartleb*, 300 F.2d 942, 133 USPQ 204 (CCPA 1962), and *Hupp v. Siroflex of America Inc.*, 122 F.3d 1456, 43 USPQ2d 1887 (Fed. Cir. 1997), which states that the reference must be identical in all material aspects.
- 10 MPEP Sec. 2131, under the title TO ANTICIPATE A CLAIM, THE REFERENCE MUST TEACH EVERY ELEMENT OF THE CLAIM, states that "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). "The identical invention must be shown in as complete 15 detail as is contained in the ... claim." *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).

Office Action fails to show the identical invention in all material aspects in as complete detail in Chow reference as is contained in the independent claims 1, 10 and 19. Therefore, it is clear that 20 the standard for determining novelty under 35 USC 102 to establish *prima facie* case of anticipation has not been met for the claimed invention, because the cited reference does not have each and every element of independent claims but have none, is from a different field, has different components, works in a different mode of use and produces different results. Thus, independent claims 1, 10, and 19 and all claims dependent upon them in the present invention 25 recite novel structure and therefore distinguish over the cited prior art, Chow, and are not anticipated by it under 35 U.S.C. 102(e).

With respect to claims 2, and 11, Office Action stated that Chow teaches comprising repeating steps (a) and (b) for remaining operation codes in the access plan (repeating the process with the

SQL query statements with loop statement each time the function statement is called: Col. 10, lines 12-18 and col. 12, lines 10-22).

5 This is also a misinterpretation because Chow does not teach comprising repeating steps (a) and (b) for remaining operation codes in the access plan, because this language is taken from the present invention. As argued above, the cited sections are directed to the WHILE statement of a SQL query.

10 Therefore, it is clear that the standard for determining novelty under 35 USC 102 to establish prima facie case of anticipation has not been met for the claimed invention, because the cited reference does not have each and every element of independent claims and claims 2 and 11 but have none, is from a different field, has different components, works in a different mode of use and produces different results. Thus, dependent claims 2 and 11 of the present invention recite novel structure and therefore distinguish over the cited prior art, Chow, and are not anticipated 15 by it under 35 U.S.C. 102(e).

20 With respect to claims 3,12 and 20, Office action stated that Chow teaches wherein said step (b) comprises augmenting said first operation code in the access plan with a pointer to an intermediate function, said intermediate function including a data structure for storing a pointer to said executable function (replacing these opcode by a lookup function to look referencing variables storing in the symbol table as the value of addressing or pointers: fig. 4, col. 24, lines 36-67 and col. 25, lines 1-8).

25 This is also a misinterpretation because the language is taken from the present invention. Moreover, Chow does not teach use of pointers to the op. code but to the variables, as held in the Office action.

30 Therefore, it is clear that the standard for determining novelty under 35 USC 102 to establish prima facie case of anticipation has not been met for the claimed invention, because the cited reference does not have each and every element of independent claims and claims 3, 12 and 20 but have none, is from a different field, has different components, works in a different mode of

use and produces different results. Thus, dependent claims 3, 12, and 20 of the present invention recite novel structure and therefore distinguish over the cited prior art, Chow, and are not anticipated by it under 35 U.S.C. 102(e).

5 With respect to claim 4, 13 and 21, Office Action stated that Chow teaches wherein said data structure includes means for storing information associated with said executable function for said first operation code (fig. 4, symbol table, item 130 & 119, col. 24, lines 36-67).

This is also a misinterpretation because the language is taken from the present invention.

10 Moreover, Chow does not teach use of pointers to the op. code but to the variables, as held in the Office action.

Therefore, it is clear that the standard for determining novelty under 35 USC 102 to establish prima facie case of anticipation has not been met for the claimed invention, because the cited 15 reference does not have each and every element of independent claims and claims 4, 13 and 21 but have none, is from a different field, has different components, works in a different mode of use and produces different results. Thus, dependent claims 4, 13, and 21 of the present invention recite novel structure and therefore distinguish over the cited prior art, Chow, and are not anticipated by it under 35 U.S.C. 102(e).

20 With respect to claim 5, 14 and 22, Office Action stated that Chow teaches wherein said step (b) comprises augmenting said first operation code in the access plan with a second pointer to a data structure, said data structure providing means for storing information associated with said first operation code or said executable function (in the symbol table has more than one local 25 variables, each for each SQL query plan, thus there is a second pointer associated with the local variable or operation code: fig. 4, col. 24, lines 36-67).

This is also a misinterpretation because the language is taken from the present invention.

Moreover, Chow does not teach use of pointers to the op. code but to the variables, as held in the 30 Office action.

Therefore, it is clear that the standard for determining novelty under 35 USC 102 to establish prima facie case of anticipation has not been met for the claimed invention, because the cited reference does not have each and every element of independent claims and claims 5, 14 and 22 but have none, is from a different field, has different components, works in a different mode of use and produces different results. Thus, dependent claims 5, 14, and 22 of the present invention recite novel structure and therefore distinguish over the cited prior art, Chow, and are not anticipated by it under 35 U.S.C. 102(e).

10 With respect to claim 6 and 15, Office action stated that Chow teaches wherein said step (a) further includes assessing the executable function associated with the first operation code and if applicable, replacing the call to the executable function with a call to a second executable (call statement invoking the SQL statement and replacing the data statement: col. 14, lines 38-67 and col. 15, lines 1-67).

15 This is also a misinterpretation because the language is taken from the present invention. Moreover, sections in col. 14, lines 38-67 and col. 15, lines 1-67 are directed to pre-processing of the control statements from a SQL query and not to the preprocessing of an existing access plan and its op. codes and functions.

20 Therefore, it is clear that the standard for determining novelty under 35 USC 102 to establish prima facie case of anticipation has not been met for the claimed invention, because the cited reference does not have each and every element of independent claims and claims 6 and 15 but have none, is from a different field, has different components, works in a different mode of use and produces different results. Thus, dependent claims 6 and 15 of the present invention recite novel structure and therefore distinguish over the cited prior art, Chow, and are not anticipated by it under 35 U.S.C. 102(e).

With respect to claims 7 and 16, Office Action stated that Chow teaches wherein said
30 intermediate function includes processing operations for the first operation code or the

executable function associated with the first operation code (col. 23, lines 45-67 and col. 24, lines 1-42).

This is also a misinterpretation because the language is taken from the present invention.

5 Moreover, sections in col. 23, lines 45-67 and col. 24, lines 1-42 are directed to a compilation and not to op. codes of an existing access plan.

Therefore, it is clear that the standard for determining novelty under 35 USC 102 to establish prima facie case of anticipation has not been met for the claimed invention, because the cited 10 reference does not have each and every element of independent claims and claims 7 and 16 but have none, is from a different field, has different components, works in a different mode of use and produces different results. Thus, dependent claims 7 and 16 of the present invention recite novel structure and therefore distinguish over the cited prior art, Chow, and are not anticipated by it under 35 U.S.C. 102(e).

15 With respect to claims 8 and 17, Office Action stated that Chow teaches wherein said processing operations in the intermediate function include gathering statistics on the user of the executable function associated with the operation code (fig. 1, item 301, analyzer generating information for the control flow and scopes and symbol table such as statistical information: col. 18, lines 20 12-40).

This is also a misinterpretation because the language is taken from the present invention. Moreover, the cited sections and Fig. 1 are not directed to op. codes of an existing access plan, symbol tables defines variables and not op. codes, and the claims are not directed to "control 25 flow and scopes".

Therefore, it is clear that the standard for determining novelty under 35 USC 102 to establish prima facie case of anticipation has not been met for the claimed invention, because the cited reference does not have each and every element of independent claims and claims 8 and 17 but 30 have none, is from a different field, has different components, works in a different mode of use

and produces different results. Thus, dependent claims 8 and 17 of the present invention recite novel structure and therefore distinguish over the cited prior art, Chow, and are not anticipated by it under 35 U.S.C. 102(e).

5 Therefore, claims 1-8, 10-17 and 19-22 are not anticipated by Chow reference under 35 U.S.C. 102(e).

7-9. Claims 9 and 18 stand rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent No.: 5,875,334 issued to Chow et al. (hereinafter Chow) in view of US Patent No.: 6,077,312 10 issued to Bates et al. ("Bates").

With respect to claims 9 and 18, Office Action stated that Chow teaches a method as discussed in claim 1, and that Chow teaches a SQL query statement is processed through a query compiler for generating a access executable plan for a database system such as IBM's DB2; this process 15 requiring a run-time interpreter and for looking up local variable; also this executable plan includes a plurality of SQL3 Query control statement for parsing into some operation codes from query graph model (QGM) representation of the statement to produce an operation code based on access executable plan, replacing the executable function in the run-time interpreter with a compiler generated local variable for looking up the symbol table with referencing local 20 variables or pointers.

As shown above, it is noted with appreciation that Office Action itself stated, on top of p. 4, that Chow is directed to pre-processing of a query, by a compiler, for generating an executable plan. Moreover, it states that interpreter there is looking up variables. Therefore, Office Action held 25 that Chow does not teach pre-processing of an access plan but of a query, before an access plan is even made, and a direct call is not even mentioned. Therefore, as proven above, the independent claims 1, 10 and 19 of the present invention are not anticipated by Chow.

It is noted with appreciation that Office Action held that Chow does not clearly teach a pause for 30 receiving user input before or after the call to the executable function.

Office Action, however, stated that Bates teaches halt execution of a computer program when the computer meets a predetermined criteria (abstract) and that, therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Chow the teachings of Bates, and that one having ordinary skill in the art would

5 have found it motivated to utilize the halt execution of a program when it is needed as disclosed (Bates' abstract), into the system of Chow for the purpose of including a pause for executing a program to replace opcode, thereby, helping to locate and identify errors in a program under development (Bates' col. 1, lines 10-20).

10 Claims 9 and 18 are directed to inserting a deliberate pause for receiving user input before or after the call to the executable function. It is accepted with gratitude that Office Action itself held that Bates teaches to halt execution of a computer program when the computer meets a predetermined criteria. Thus, Office Action itself held that in Bates it is the decision of the computer program when to stop, i.e., when the computer meets a predetermined criteria, and not 15 of the user. It also held that it halts (permanently stops) and does not just make a pause in order to receive the user's input. Moreover, Bates reference is directed to debugging utilizing a context sensitive breakpoints, which is a completely different field.

Further, Applicant challenges the Office Action "would have been obvious" allegation used to 20 reject claims 9 and 18 of the present invention under Sec. 103 and, as allowed under MPEP Sec. 2144.03, respectfully requests that Examiner cites prior art references which support all these "would have been obvious" allegations and show how modifications can be accomplished in the cited reference and what motivation was used to modify a reference to arrive at the claimed subject matter and to show how this combination of modified references functions and which 25 structure it has and how it makes the claimed present invention obvious.

Further, Bates is motivated to halt the execution of a program when it is needed to locate and identify errors in a program under development (Bates' col. 1, lines 10-20). However, the present invention is not directed to debugging so the motivation given in the Office Action to combine 30 these references is clearly wrong.

Moreover, Applicant includes herein the same argument as regarding claims 1, 10 and 19 because Bates reference shows a different system and method from the present invention and so does the Chow reference, as shown above, and neither reference is directed to the direct call 5 threaded code of the present invention. Therefore, the references cannot be combined and be used to reject claims 9 and 18, which recite novel and unobvious structure and use and, therefore, distinguish over the cited prior art, Chow and Bates.

Further, the Examiner has not established a *prima facie* case of obviousness because the three 10 basic criteria, which must be met, were not met. Office Action has no suggestion or motivation, either in the reference or in the knowledge generally available to one of ordinary skill in the art to modify the reference or to combine reference teachings, a reasonable expectation of success was not shown (and is impossible) and the prior art reference(s), which must teach or suggest all the claim limitations, does not do so here. Furthermore, the Examiner did not satisfy the initial 15 burden to provide some suggestion in the references of the desirability of doing what the inventor has done, because to support the conclusion that the claimed invention is directed to obvious subject matter, either the references must expressly or impliedly suggest the claimed invention or the Examiner must present a convincing line of reasoning as to why the artisan would have found the claimed invention to have been obvious in light of the teachings of the 20 references. Furthermore, the cited reference is from nonanalogous art, under MPEP Sec. 2141.01(a).

These references are from completely different fields unrelated to a pre-processing of access 25 plans using pointers and the methods taught by these two references are among themselves from completely different art fields and cannot be combined. Moreover, it is shown that they do not perform any elements of the independent claims 1, 10, and 19 and therefore their dependent claims. Further, they satisfy a different need from a different area and do not teach pre-processing of existing access plan. Therefore, these references cannot be used to invalidate independent claims 1, 10, and 19 and their dependent claims. Because none of the referenced 30 prior art teaches elements (a) to (b) of claims 1, 10 and 19, which are the main steps of the

present invention, their combination is not a valid reason for rejection of these independent claims and claims dependent thereof. Therefore, each cited reference, by itself or in combination, cannot be used to invalidate claims 1, 10 and 19 because they fail to teach any and all the steps of these claims.

5

Because the referenced prior art does not teach elements of claims 1, 10, and 19 and their dependent claims 9 and 18, there is no valid reason for rejection of these independent claims and claims dependent thereof because the cited references cannot be used to invalidate these claims and, thus, a *prima facie* case of obviousness has not been established under 35 USC Sec. 103(a).

10

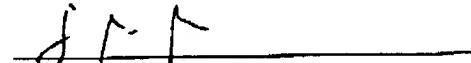
Regarding claims 1-22, none of the cited references teaches, shows, suggests or is even remotely related to pre-processing of existing access plans to augment op. code with pointers, as claimed by the present invention. Therefore, these references cannot be used to invalidate independent claims 1, 10, and 19 and their dependent claims. Moreover, the Examiner combined references from different arts in order to reject claims 1-22, by quoting parts of sentences nonexistent in those references. However, even if these quotes are correct, the combination must be pointed to in the prior art itself and no such combination is pointed to in the cited references nor it could be since they are from different fields. Therefore, these references cannot be used to invalidate independent claims 1, 10 and 19 and their dependent claims because they fail to teach any and all the steps of these claims.

Therefore, all submitted claims are allowable over the cited references and their reconsideration is respectfully requested. Improper combination of cited references is used in each claim rejection in the Office Action. None of the cited references suggests combination under *In re Sernaker*, 217 U.S.P.Q. 1, 6 (CAFC 1983), and one skilled in the art would have no reason to make a combination since they are from different fields, impossible to combine and individually complete. Moreover, none of the cited references discloses the subject matter and features of claims 1-22 of the present invention and even if they did show some individual features, they would not be able to meet the claims of the present invention which provide new and unexpected

results over these references and are thus novel and unobvious and patentable under Sections 102 and 103.

In view of the above, it is submitted that this application is now in good order for allowance.
5 which applicant respectfully solicits. Should matters remain which the Examiner believes could be resolved in a telephone interview, the Examiner is kindly requested to telephone the Applicant's undersigned attorney. No additional fee is required in connection with this communication since the Response is mailed within three months from the Office Action and the number of claims is not extending the original number of claims. However, any underpayment is
10 authorized to be charged to Deposit Account Number 09-0460 in the name of IBM Corporation.

Respectfully submitted,


Sandra M. Parker
Reg. No. 36,233

LAW OFFICE OF SANDRA M. PARKER
329 La Jolla Avenue
Long Beach, CA 90803
Phone/Fax: (562) 597-7504

15

20